View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Elsevier - Publisher Connector

Discrete Mathematics 70 (1988) 137–147
North-Holland

A POLYNOMIAL ALGORITHM FOR CONSTRUCTING FAMILIES OF k-INDEPENDENT SETS

G. FREIMAN, E. LIPKIN
School of Mathematical Sciences, Tel Aviv University, Ramat Aviv, Tel Aviv, Israel

L. LEVITIN
College of Engineering, Boston University, Boston, MA 02215, U.S.A.

Received September 9, 1986
Revised April 2, 1987

The present paper describes an algorithm for constructing families of k-independent subsets F_k of $\{1, 2, \ldots, n\}$ with $|F_k| \geq 2^{c_k n}$, where $c_k = d/(k-1)2^k$ and d is a certain constant. The algorithm has a polynomial complexity with respect to the size of the family constructed.

1. Introduction

The family F of subsets of $N = \{1, 2, \ldots, n\}$ is called k-independent, if for every k distinct subsets $X_1, \ldots, X_k$ of F all 2^k intersections $\bigcap_{j=1}^{k} Y_j$ are non-empty, where Y_j can be either X_j, or its complement $\bar{X}_j$. Kleitman and Spencer [1] have proved, that for every fixed k there exists a k-independent family F_k on n elements of maximal size $|F_k| = f(n, k)$, where

$$2^{d_1(n/k2^k)} \leq f(n, k) \leq 2^{d_2(n/2^k)}, \tag{1}$$

and d_1, d_2 are absolute constants.

Alon [2] and Levitin, Karpovsky [3], using the theory of error correcting codes, give explicit constructions of k-independent families; [2] gives $|F_k| > 2^{c_k n}$, where $c_k = [\log(7k^4)]/[2^{135k^4}(135k^4)^{24k^2}]$, [3] base their explicit construction of MDS codes, which gives asymptotically exponential size of F_k.

The present paper describes a method of constructing k-independent families F_k of size $|F_k| > 2^{c_k n}$, where $c_k = d/(k-1)2^k$ and $d = \frac{1}{5}$. For the special case $k = 3$ we obtain a substantial improvement of Kleitman–Spencer's lower bound.

2. Notations

Each subset $X_j \subseteq N = \{1, 2, \ldots, n\}$ of l elements ($|X_j| = l$) is uniquely described by its characteristic column-vector which has l ones and $(n-l)$ zeros. Henceforth we denote the characteristic vector of X_j by the same symbol j, since the subsets themselves will not be used below. Thus the family of subsets

0012-365X/88/$3.50 © 1988, Elsevier Science Publishers B.V. (North-Holland)

$F = \{X_1, \ldots, X_q\}$ corresponds to a binary matrix $M(n, q)$ of size $n \times q$, which has the characteristic vectors $X_1, \ldots, X_q$ as columns. The family F (with $|F| \geq k$) is k-independent, if each submatrix $M(n, k)$ of its matrix contains all 2^k different rows (i.e., all the possible binary vectors of length k).

Denote by $r_{a_1,\ldots,a_\nu}$ the number of rows $(a_1, \ldots, a_\nu)$ in a ν-tuple of columns $[X_{i_1}, \ldots, X_{i_\nu}]$, where $a_\mu \in \{0, 1\}$; $\mu = 1, \ldots, \nu$; $\nu = 1, \ldots, k-1$. We call a ν-tuple admissible, if $r_{a_1,\ldots,a_\nu} > \alpha_\nu n - 1$, where $\alpha_\nu > 0$ $(\nu = 2, \ldots, k-1)$ are some fixed parameters.

3. The case $k = 3$

3.1. *Description of the algorithm*

To explain in a simple way the basic ideas of the algorithm, let us first consider the case $k = 3$. (Moreover, this case has been investigated in more details, and an improvement of the lower bound given in [1] has been obtained.) We will construct a 3-independent family F_3 by recursion. At the $(s+1)$th step of the algorithm we add one column to the set $F_3^{(s)}$ of columns obtained before, the column added at the step $(s+1)$ being chosen from a set $T^{(s)}$ determined at the previous steps of the algorithm and called the "resource". We denote by $A_j^{(s)}$, $j = 1, \ldots, s$ the columns belonging to $F_3^{(s)}$, and by $B_j^{(s)}$, $j = 1, \ldots, |T^{(s)}|$ the columns belonging to $T^{(s)}$.

In the beginning $F^{(0)} = \emptyset$, $T^{(0)}$ is the largest possible family F_2 of 2-independent columns of length n. As shown in [1], F_2 consists of all columns with $r_0 = \lfloor \frac{1}{2}n \rfloor$ and such that they all have zero in the same position (e.g., in the first one). Therefore

$$|T^{(0)}| = |F_2| = \binom{n-1}{\lfloor \frac{1}{2}n \rfloor - 1}. \tag{2}$$

(Here $\lfloor y \rfloor$ is the integral part of y.)

At the first step of the algorithm we form $F^{(1)} = \{A_1^{(1)}\}$ by choosing an arbitrary column $A_1^{(1)} \in T^{(0)}$. Then we "weed" the remaining set of columns $T^{(0)} - \{A_1^{(1)}\}$ discarding all the vectors $B_j^{(0)} \in T^{(0)}$ such that $[A_1^{(1)}, B_j^{(0)}]$ is not an admissible pair. The set of columns survived after this procedure is $T^{(1)}$.

At the second step of the algorithm we form $F^{(2)} = \{A_1^{(2)}, A_2^{(2)}\}$, where $A_1^{(2)} = A_2^{(2)}$ and $A_2^{(2)}$ is an arbitrary vector belonging to $T^{(1)}$. Then we weed the set $T^{(1)} - \{A_2^{(2)}\}$ in the following way:

(i) we discard all $B_j^{(1)} \in T^{(1)}$ such that the pairs $[A_2^{(2)}, B_j^{(1)}]$ are not admissible;

(ii) we discard all $B_j^{(1)} \in T^{(1)}$ such that the triplets $[A_1^{(2)}, A_2^{(2)}, B_j^{(1)}]$ are not 3-independent.

The remaining columns form the resource $T^{(2)}$.

Suppose that after s steps $(s \geq 2)$ we have a family $F^{(s)} = |A_1^{(s)}, \ldots, A_s^{(s)}]$ and a

"store" $T^{(s)}$ that has been constructed to have the following properties: for $1 \leq i_1, i_2 \leq s$

(i) all pairs $[A_{i_1}^{(s)}, B_j^{(s)}]$ are admissible;

(ii) all 3-tuples $[A_{i_1}^{(s)}, A_{i_2}^{(s)}, B_j^{(s)}]$ are 3-independent.

At the $(s+1)$th step of the algorithm we form $F^{(s+1)} = \{A_1^{(s+1)}, \ldots, A_{s+1}^{(s+1)}\}$, where $A_i^{(s+1)} = A_i^{(s)}$, $i = 1, \ldots, s$ and $A_{s+1}^{(s+1)} \in T^{(s)}$ is chosen arbitrarily. Then we prepare the resource $T^{(s+1)}$ in the following way:

(i) we discard all $B_j^{(s)} \in T^{(s)}$ such that the pairs $[A_{s+1}^{(s+1)}, B_j^{(s)}]$ are not admissible;

(ii) we discard all $B_j^{(s)}$ such that at least one of the triplets $[A_i^{(s+1)}, A_{s+1}^{(s+1)}, B_j^{(s)}]$ is not admissible $(i = 1, \ldots, s)$.

The algorithm terminates if $T^{(s+1)} = \emptyset$.

3.2. *Estimation of the size of the 3-independent family constructed*

Since we are interested in the estimation of the size of F_3 for large n we assume that n is even. (If n is odd we can replace n by $n-1$, which will not change the asymptotic results).

Denote by $u(s)$ the number of columns discarded from the resource at the sth step of the algorithm. Then

$$u(s) = u_2(s) + u_3(s), \tag{3}$$

where $u_2(s)$ and $u_3(s)$ are the numbers of columns discarded at the stages (i) and (ii) of the step, respectively.

Obviously,

$$u_2(s) \leq \sum_{a_1, a_2} u_{a_1 a_2}(s), \tag{4}$$

where $a_1, a_2 \in \{0, 1\}$ and $u_{a_1 a_2}$ is the number of columns $B_j^{(s-1)} \in T^{(s-1)}$ such that $r_{a_1, a_2} \leq \lfloor \alpha n \rfloor - 1$ in the pair $[A_s^{(s)}, B_j^{(s-1)}]$. $u_{a_1 a_2}(s)$ can be upper bounded as follows:

$$u_{a_1 a_2}(s) \leq \sum_{i=0}^{\lfloor \alpha n \rfloor - 1} \binom{\frac{1}{2}n}{i}\binom{\frac{1}{2}n}{\frac{1}{2}n - i} = \sum_{i=0}^{\lfloor \alpha n \rfloor - 1} \binom{\frac{1}{2}n}{i}^2$$

$$< \left[\sum_{i=0}^{\lfloor \alpha n \rfloor - 1} \binom{\frac{1}{2}n}{i}\right]^2 \leq 2^{nh(2\alpha)}, \tag{5}$$

where $h(y) = -y \log_2 y - (1-y)\log_2(1-y)$ is the binary entropy function. Thus

$$u_2(s) < 4 \cdot 2^{nh(2\alpha)}. \tag{6}$$

Let us now estimate $u_3(s)$, i.e., the number of non-3-independent triplets $[A_i^{(s)}, A_s^{(s)}, B_j^{(s-1)}]$.

$$u_3(s) \leq \sum_{i=1}^{s-1} \sum_{a_1, a_2, a_3} u_{a_1, a_2, a_3}(A_i^{(s)}), \tag{7}$$

where $a_1, a_2, a_3 \in \{0, 1\}$ and $u_{a_1a_2a_3}(A_i^{(s)})$ is the number of triplets $[A_i^{(s)}, A_s^{(s)}, B_j^{(s-1)}]$ in which $r_{a_1a_2a_3} = 0$. But $u_{a_1a_2a_3}(A_i^{(s)})$ is upper-bounded by

$$u_{a_1a_2a_3}(A_j^{(s)}) \leqslant \binom{n - r_{a_1a_2}}{\frac{1}{2}n} \leqslant \binom{n - \lfloor \alpha n \rfloor}{\frac{1}{2}n} \leqslant 2^{n(1-\alpha)h(1/2(1-\alpha))}. \tag{8}$$

Thus

$$u_s(s) < 8(s-1)2^{n(1-\alpha)h(1/2(1-\alpha))}. \tag{9}$$

Suppose that the algorithm terminates after N steps, so that $|F_3| = N$. Then the total number of discarded columns

$$u = \sum_{s=1}^{N} u(s) < N \cdot 2^{nh(2\alpha)+2} + N^2 \cdot 2^{n(1-\alpha)h(1/2(1-\alpha))+2}. \tag{10}$$

The following condition should hold:

$$N + u = |T^{(0)}| = \binom{n-1}{\frac{1}{2}n - 1}. \tag{11}$$

Therefore by (10), a lower bound $\hat{N}$ on N can be obtained from the inequality

$$\hat{N}(1 + 2^{nh(2\alpha)+2}) + \hat{N}^2 \cdot 2^{n(1-\alpha)h(1/2(1-\alpha))+2} \leqslant \binom{n-1}{\frac{1}{2}n - 1}. \tag{12}$$

To satisfy (12), each term in the left-hand part of (12) should not exceed the right-hand part. Therefore,

$$\hat{N} \leqslant \min\left\{\binom{n-1}{\frac{1}{2}n-1}2^{-nh(2\alpha)}, \binom{n-1}{\frac{1}{2}n-1}^{\frac{1}{2}} 2^{-(1-\alpha)\frac{n}{2}h(1/2(1-\alpha))}\right\}. \tag{13}$$

On the other hand, (12) is satisfied, if each term at the left-hand side does not exceed one half of the right-hand part. Therefore, a sufficient condition on $\hat{N}$ to be a lower bound is given by

$$\hat{N} = \min\left\{\frac{1}{2}\binom{n-1}{\frac{1}{2}n-1}[1 + 2^{nh(2\alpha)+2}]^{-1}, \binom{n-1}{\frac{1}{2}n-1}^{\frac{1}{2}} 2^{-(1-\alpha)\frac{n}{2}h(1/2(1-\alpha))-2}\right\}. \tag{14}$$

The maximum value of $\hat{N}$ which satisfies (13) or (14) is obtained for such value of α that both functions in the braces are equal, since in both (13) and (14) one of these functions increases and the other decreases with α. But both (13) and (14) give the same equation for α with the accuracy of terms of order $O((\ln n/n))$. The equation has the following form:

$$1 - 2h(2\alpha) + (1-\alpha)h\left(\frac{1}{2(1-\alpha)}\right) = 0. \tag{15}$$

which gives

$$\alpha = 0.1615, \tag{16}$$

Substituting the value (16) for α in (13) or (14), we obtain:

$$N \geqslant \hat{N} \geqslant 2^{(c_3-\varepsilon)n}, \tag{17}$$

where

$$c_3 = 1 - h(2\alpha) = 0.0920 \tag{18}$$

and $\varepsilon > 0$ is arbitrarily small.

(Here $f(y) \geqslant g(y)$ means, as usual, that $\lim_{y\to\infty} f(y)/g(y) \geqslant 1$; $f(y) \sim g(y)$ means that both $f(y) \geqslant g(y)$ and $g(y) \geqslant f(y)$.)

The lower bound on $f(n, 3)$ obtained in [1] is

$$f(n, s) \geqslant 2^{0.0642n(1+o(1))}.$$

Thus our constant c_3 gives a considerable improvement of the lower bound.

4. The case of arbitrary k

4.1. *Description of the algorithm*

The algorithm for the general case of any $k \leqslant \lfloor \log n \rfloor$ is quite similar to that for $k = 3$. The major difference is that instead of one parameter α we introduce a set of parameters $\{\alpha_\nu\}$, $\nu = 2, \ldots, k-1$. The parameter α_ν determines whether a given ν-tuple of columns is admissible, as defined in Section 2. We also require that

$$\alpha_\nu < 2^{-\nu} \quad \text{and} \quad \alpha_{\nu+1} < \tfrac{1}{2}\alpha_\nu. \tag{19}$$

Denote by $F_k^{(s)}$ and $T_k^{(s)}$ the family of k-independent columns and, respectively, the resource formed at the sth step of the algorithm. In the beginning we take $F_k^{(0)} = \emptyset$ and $T_k^{(0)} = F_2$. Suppose now that after s steps of the algorithm we have a resource $T_k^{(s)}$ and a family $F_k^{(s)} = \{A_1^{(s)}, \ldots, A_s^{(s)}\}$ of columns having the following properties:

(a) all ν-tuples $[A_{i_1}^{(s)}, \ldots, A_{i_\nu}^{(s)}]$ $(\nu = 2, \ldots, k-1, i_1, \ldots, i_\nu \in \{1, \ldots, s\})$ are admissible;

(b) all k-tuples $[A_{i_1}^{(s)}, \ldots, A_{i_k}^{(s)}]$ $(i_1, \ldots, i_k \in \{1, \ldots, s\})$ are k-independent.

(The empty set of ν-tuples or k-tuples is admissible by definition). At the $(s+1)$th step of the algorithm, we form $F_k^{(s+1)} = \{A_1^{(s+1)}, \ldots, A_{s+1}^{(s+1)}\}$, where $A_i^{(s+1)} = A_i^{(s)}$, $i = 1, \ldots, s$ and $A_{s+1}^{(s+1)} \in T_k^{(s)}$ is chosen arbitrarily. Then we prepare the next resource $T_k^{(s+1)}$ in the following way:

(1) we discard all $B_j^{(s)} \in T^{(s)}$ such that at least one of the ν-tuples $[A_{i_1}^{(s+1)}, \ldots, A_{i_{\nu-2}}^{(s+1)}, A_{s+1}^{(s+1)}, B_j^{(s)}]$ is not admissible $(\nu = 2, \ldots, k-1;$ $i_1, \ldots, i_{\nu-2} \in \{1, \ldots, s\})$;

(2) we discard all $B_j^{(s)} \in T^{(s)}$ such that at least one of the k-tuples $[A_{i_1}^{(s+1)}, \ldots, A_{s+1}^{(s+1)}, B_j^{(s)}]$ is not admissible $(i_1, \ldots, i_{k-2} \in \{1, \ldots, s\})$.

Obviously, the set $F_k^{(s+1)}$ retains the properties (a) and (b) (with $(s+1)$ substituted for s).

The algorithm terminates if $T_k^{(s+1)} = \emptyset$.

4.2. *Estimation of the size of the k-independent family*

Let us estimate the size of the k-independent family constructed by the algorithm described above. We assume that k is constant, n is even, and estimate the asymptotic growth of $|F_k|$ for $n \to \infty$.

The number of columns discarded from the resource at the sth step of the algorithm is

$$u(s) = \sum_{\nu=2}^{k-1} u_\nu(s) + u_k(s), \tag{20}$$

where $u_\nu(s)$ is the number of columns discarded because they form inadmissible ν-tuples and $u_k(s)$ is the number of columns discarded because they form non-k-independent k-tuples. Obviously,

$$u_\nu(s) \leqslant \sum_{m=1}^{\binom{s-1}{\nu-2}} \sum_{a_1,\ldots,a_\nu} u_{a_1,\ldots,a_\nu}(L_m^{(\nu-1)}), \tag{21}$$

where $a_1, \ldots, a_\nu \in \{0, 1\}$, $L_m^{(\nu-1)}$ is a $(\nu-1)$-tuple of columns $[A_{i_1}^{(s)}, \ldots, A_{i_{\nu-1}}^{(s)}, A_s^{(s)}] \subseteq F_k^{(s)}$ and $u_{a_1,\ldots,a_\nu}(L_m^{(\nu-1)})$ is the number of columns from $T_k^{(s-1)}$ such that they form inadmissible ν-tuples with a given $(\nu-1)$-tuple $L_m^{(\nu-1)}$.

Consider a column $B_j^{(s-1)} \in T_k^{(s-1)}$. The column is not admissible with respect to a row $(a_1, \ldots, a_{\nu-1}, a_\nu)$ if it contains less than $\lfloor \alpha_\nu n \rfloor$ digits of the type α_ν in those positions, where the $(\nu-1)$-tuple $L_m^{(\nu-1)}$ contains rows $(a_1, \ldots, a_{\nu-1})$. Note that the total number of positions occupied by a_ν is $\frac{1}{2}n$. Hence

$$\begin{aligned} u_{a_1,\ldots,a_\nu}(L_m^{(\nu-1)}) &\leqslant \sum_{i=0}^{\lfloor \alpha_\nu n \rfloor - 1} \binom{r_{a_1,\ldots,a_{\nu-1}}}{i} \binom{n - r_{a_1,\ldots,a_{\nu-1}}}{\frac{1}{2}n - i} \\ &\leqslant \sum_{i=0}^{\lfloor \alpha_\nu n \rfloor - 1} \binom{\lfloor \alpha_{\nu-1} n \rfloor}{i} \binom{n - \lfloor \alpha_{\nu-1} n \rfloor}{\frac{1}{2}n - i} \\ &< \sum_{i=0}^{\lfloor \alpha_\nu n \rfloor - 1} \binom{\lfloor \alpha_{\nu-1} n \rfloor}{i} \cdot \sum_{j=0}^{\lfloor \alpha_\nu n \rfloor - 1} \binom{n - \lfloor \alpha_{\nu-1} n \rfloor}{\frac{1}{2}n - j} \\ &\leqslant 2^{\alpha_{\nu-1} n h(\alpha_\nu/\alpha_{\nu-1}) + (1-\alpha_{\nu-1}) n h(1 - 2\alpha_\nu/2(1-\alpha_{\nu-1})) + 1}. \end{aligned} \tag{22}$$

(Here, by definition, $\alpha_1 = \frac{1}{2}$, which agrees with the definition of $T_k^{(0)}$.)

Similarly

$$u_k(s) \leqslant \sum_{m=0}^{\binom{s-1}{k-2}} u_{a_1,\ldots,a_k}(L_m^{(k-1)}), \tag{23}$$

where $L_m^{(k-1)} = [A_{i_1}^{(s)}, \ldots, A_{i_{k-2}}^{(s)}, A_s^{(s)}] \subseteq F_k^{(s)}$, and $u_{a_1,\ldots a_k}(L_m^{(k-1)})$ is the number of

columns from $T_k^{(s-1)}$ such that they form non-k-independent k-tuples with a given $(k-1)$-tuple $L_m^{(k-1)}$.

The upper bound for $u_{a_1,\dots,a_k}(L_m^{(k-1)})$ is given by

$$u_{a_1,\dots,a_k}(L_m^{(k-1)}) \leqslant \binom{n - r_{a_1,\dots a_{k-1}}}{\frac{1}{2}n} \leqslant \binom{n - \lfloor \alpha_{k-1} n \rfloor}{\frac{1}{2}n} < 2^{n(1-\alpha_{k-1}h(1/2(1-\alpha_{k-1}))}. \tag{24}$$

Thus

$$u(s) < \sum_{\nu=2}^{k-1} 2^{\nu} \binom{s-1}{\nu-2} \cdot 2^{\alpha_{\nu-1} n h(\alpha_\nu/\alpha_{\nu-1}) + (1-\alpha_{\nu-1}) n h((1-2\alpha_\nu)/2(1-\alpha_{\nu-1}))+1} + 2^k \binom{s-1}{k-2} \cdot 2^{n(1-\alpha_{k-1})h(1/2(1-\alpha_{k-1}))}. \tag{25}$$

Suppose that the algorithm terminates after N_k steps. Taking into account that

$$\sum_{s=1}^{N_k} \binom{s-1}{\nu-2} = \binom{N_k}{\nu-1}$$

and $2^{\nu-2} < (\nu-1)!$ for $\nu \geqslant 2$ we obtain the following upper bound for the total number of discarded columns:

$$u = \sum_{\nu=1}^{N} u(s) < \sum_{\nu=2}^{k-1} N_k^{\nu-1} \cdot 2^{\alpha_{\nu-1} n h(\alpha_\nu/\alpha_{\nu-1}) + (1-\alpha_{\nu-1}) n h((1-2\alpha_\nu)/2(1-\alpha_{\nu-1}))+3} + N_k^{k-1} 2^{n(1-\alpha_{k-1})h(1/2(1-\alpha_{k-1}))+2}. \tag{26}$$

Since

$$N_k + U = |T^{(0)}| = \binom{n-1}{\frac{1}{2}n-1},$$

the lower bound $\hat{N}_k$ on N satisfies the inequality:

$$\binom{n-1}{\frac{1}{2}n-1} \geqslant \hat{N}_k + \sum_{\nu=2}^{k-1} \hat{N}_k^{\nu-1} \cdot 2^{\alpha_{\nu-1} n h(\alpha_\nu/\alpha_{\nu-1}) + (1-\alpha_{\nu-1}) n h((1-2\alpha_\nu)/2(1-\alpha_{\nu-1}))+3} + \hat{N}_k^{k-1} 2^{n(1-\alpha_{k-1})h(1/2(1-\alpha_{k-1}))+2} \tag{27}$$

By the same reasoning as in the case $k = 3$ we obtain the following conditions that $\hat{N}$ satisfies (27): the necessary condition

$$\hat{N}_k \leqslant \min\left\{ \binom{n-1}{\frac{1}{2}n-1}^{\frac{1}{\nu-1}} 2^{-n\frac{\alpha_{\nu-1}}{\nu-1} h\left(\frac{\alpha_\nu}{\alpha_{\nu-1}}\right) - n\frac{1-\alpha_{\nu-1}}{\nu-1} h\left(\frac{1-2\alpha_\nu}{2(1-\alpha_{\nu-1})}\right)}, \binom{n-1}{\frac{1}{2}n-1}^{\frac{1}{k-1}} \cdot 2^{-n\frac{1-\alpha_{k-1}}{k-1} h\left(\frac{1}{2(1-\alpha_{k-1})}\right)} \right\}, \quad \nu = 2, \dots, k-1; \tag{28}$$

and the sufficient condition,

$$\hat{N}_k = \min\left\{\frac{1}{k-1}\binom{n-1}{\frac{1}{2}n-1}[2^{nh(2\alpha_2)+3}+1]^{-1},\right.$$
$$\left[\frac{1}{k-1}\binom{n-1}{\frac{1}{2}n-1}2^{-\alpha_{\nu-1}nh\left(\frac{\alpha_\nu}{\alpha_{\nu-1}}\right)-(1-\alpha_{\nu-1})nh\left(\frac{1-2\alpha_\nu}{2(1-\alpha_{\nu-1})}\right)-3}\right]^{\frac{1}{\nu-1}},$$
$$\left.\left[\frac{1}{k-1}\binom{n-1}{\frac{1}{2}n-1}2^{-(1-\alpha_{k-1})nh\left(\frac{1}{2(1-\alpha_{k-1})}\right)-2}\right]^{\frac{1}{k-1}}\right\}, \quad \nu=3,\ldots,k-1. \tag{29}$$

It can be shown that the maximum value of $\hat{N}_k$ which satisfies (28) or (29) is achieved for such values of $\alpha_2,\ldots,\alpha_{k-1}$ for which all the functions in the right-hand side of inequalities (28) (or, respectively, (29)) are equal. The equations for $\alpha_2,\ldots,\alpha_{k-1}$ which follow under this condition from (28) and (29) coincide asymptotically with the accuracy of terms of order $O((\ln n)/n)$.

$$1-\alpha_{\nu-1}h\left(\frac{\alpha_\nu}{\alpha_{\nu-1}}\right)-(1-\alpha_{\nu-1})h\left(\frac{1-2\alpha_\nu}{2-2\alpha_{\nu-1}}\right)$$
$$=\frac{\nu-1}{k-1}\left[1-(1-\alpha_{k-1})h\left(\frac{1}{2-2\alpha_{k-1}}\right)\right], \quad \nu=2,\ldots,k-1. \tag{30}$$

To find a lower bound on $\hat{N}_k$ we will derive from (30) a lower bound on α_{k-1}. (Henceforth we imply that α_ν, α_{k-1} are roots of the system of Eqs. (30)).

Denote

$$\alpha_{k-1}=\frac{\beta^{(k)}}{2^{k-1}}. \tag{31}$$

Then by the use of inequality

$$h(\tfrac{1}{2}\pm y)\geq 1-4y^2\log_2 e, \tag{32}$$

we obtain

$$1-(1-\alpha_{k-1})h\left(\frac{1}{2-2\alpha_{k-1}}\right)\leq\frac{1}{2^{k-1}}\beta^{(k)}\left(1+\frac{\beta^{(k)}\log_2 e}{2^{k-1}-\beta^{(k)}}\right)=\frac{\gamma^{(k)}}{2^{k-1}}. \tag{33}$$

By the use of inequality

$$h(\tfrac{1}{2}\pm y)\leq 1-2y^2\log_2 e \tag{34}$$

we have

$$1-\alpha_{\nu-1}h\left(\frac{\alpha_\nu}{\alpha_{\nu-1}}\right)-(1-\alpha_{\nu-1})h\left(\frac{1-2\alpha_\nu}{2-2\alpha_{\nu-1}}\right)\geq\frac{\log_2 e(\alpha_{\nu-1}-2\alpha_\nu)^2}{2\alpha_{\nu-1}(1-\alpha_{\nu-1})} \tag{35}$$

Hence, from (30), (33) and (35),

$$\frac{\log_2 e(\alpha_{\nu-1}-2\alpha_\nu)^2}{2\alpha_{\nu-1}(1-\alpha_{\nu-1})}\leq\frac{(\nu-1)\gamma^{(k)}}{(k-1)2^{k-1}}. \tag{36}$$

Thus, taking into account (19), we obtain:

$$\alpha_v \geqslant \frac{\alpha_{v-1}}{2} - \sqrt{\frac{(v-1)\gamma^{(k)}\alpha_{v-1}(1-\alpha_{v-1})}{\log_2 e(k-1)2^k}} \geqslant \frac{\alpha_{v-1}}{2} - \sqrt{\frac{(v-1)\gamma^{(k)}(1-2^{-(v-1)})}{(k-1)2^{k+v-1}\log_2 e}}. \tag{37}$$

This leads to the following inequality for α_v assuming $\alpha_1 = \frac{1}{2}$:

$$\alpha_v \geqslant \frac{1}{2^v} - 2^{-(v-1)}\sqrt{\frac{\gamma^{(k)}}{(k-1)2^k \log_2 e}} \sum_{i=1}^{v-1} \sqrt{i(2^i-1)}$$

$$\geqslant \frac{1}{2^v} - 2^{-(v-1)}\sqrt{\frac{\gamma^{(k)}(v-1)(2^{v-1}-1)}{(k-1)2^k \log_2 e}}(2+\sqrt{2})\left(1 - \frac{\sqrt{2}-1}{2(v-1)}\right) \tag{38}$$

Hence, for $v = k-1$, by (31) and (38) we obtain:

$$\beta^{(k)} \geqslant 1 - \sqrt{\frac{\gamma^{(k)}(k-2)\ln 2}{k-1}}(2+\sqrt{2})\left(1 - \frac{\sqrt{2}-1}{2(k-1)}\right). \tag{39}$$

Thus, by (33) and (39), $\beta^{(k)}$ satisfies inequality

$$(1-\beta^{(k)})^2 \leqslant (2+\sqrt{2})^2 \ln 2 \cdot \left(\frac{k-2}{k-1}\right)\left(1 - \frac{\sqrt{2}-1}{2(k-2)}\right)^2 \beta^{(k)}\left(1 + \frac{\beta^{(k)}}{(2^{k-1}-\beta^{(k)})\ln 2}\right). \tag{40}$$

The left-hand part of this inequality is a decreasing function of $\beta^{(k)}$, while the right-hand part increases with $\beta^{(k)}$. Therefore the minimum value of $\beta^{(k)}$ is obtained when (40) turns into equality. If $\beta_0^{(k)}$ is the root of (40) (considered as an equation for β), then

$$\alpha_{k-1} \geqslant \frac{1}{2^{k-1}}\beta_0^{(k)}. \tag{41}$$

Moreover, it is seen that the right-hand part of (40) is an increasing function of k (for a fixed β). Therefore, $\beta_0^{(l)}$ decreases with the increase of k. Thus, for any k,

$$\alpha_{k-1} \geqslant \frac{1}{2^{k-1}}\beta_0^{(\infty)}, \tag{42}$$

where $\beta_0^{(\infty)} = \lim_{k\to\infty}\beta_0^{(k)}$ is the root of the equation obtained from (40) by setting $k = \infty$:

$$(1-\beta^{(\infty)})^2 = (2+\sqrt{2})^2 \ln 2 \cdot \beta^{(\infty)}. \tag{43}$$

This gives $\beta_0^{(\infty)} = 0.1002033\ldots$. Since $1-(1-\alpha_{k-1})h(1/(2-2\alpha_{k-1})) \geqslant \alpha_{k-1} \geqslant \beta_0^{(\infty)}/2^{k-1}$, it follows from (29) that

$$N_k \geqslant \hat{N}_k \geqslant 2^{\beta_0^{(\infty)} n[(k-1)2^{k-1}]^{-1}} > 2^{n/5(k-1)2^{-k}}. \tag{44}$$

Note that a larger constant in the exponent can be obtained for any given k by solving the equation (40). For example, $\beta_0^{(3)} = 0.1988$, $\beta_0^{(4)} = 0.1645$, $\beta_0^{(5)} = 0.1422$. (The exact solution in Section 3 gives $\beta^{(3)} = 0.72$.)

5. Complexity of the algorithm

Let us estimate the complexity of the described algorithm. The total number W_k of operations (by an operation we mean a check of a ν-tuple for admissibility of k-independence) is upperbounded by

$$\begin{aligned} W_k &= \sum_{s=1}^{N_k} \sum_{\nu=2}^{k} \binom{s-1}{\nu-2} |T^{(s-1)}| \\ &< |T^{(0)}| \sum_{s=1}^{N_k} \sum_{\nu=2}^{k} \binom{s-1}{\nu-2} \\ &= |T^{(0)}| \sum_{\nu=2}^{k} \binom{N_k}{\nu-1} \\ &\leqslant |T^{(0)}| \, 2^{N_k h((k-1)/N_k)} < 2^{n(1+5\cdot 2^{-k})}. \end{aligned}$$

Since the number of columns in F_k is bounded by (44), we obtain

$$W_k \leqslant N_k^{(k-1)(5\cdot 2^k+1)}.$$

Thus the algorithm is polynomial in N_k for any fixed k, though the degree of the polynomial grows rather fast with k. It should be mentioned that the complexity of the algorithm can be substantially reduced by successive construction of the families F_k. Indeed, we can use F_{k-1} as the initial resource for finding F_k (note that the optimal α_ν for F_{k-1} are slightly different from those for F_k). Then a rough estimation of the number of operations $W_{k|k-1}$ required to find a family F_k starting with F_{k-1} gives:

$$W_{k|k-1} \approx N_k^{k+1+2/(k-2)},$$

i.e., the degree of the polynomial grows linearly with k.

Recently the construction of k-independent families has become of practical importance: the columns of F_k form the matrix of the so called s-exhaustive test for digital circuits, the rows of the matrix being the test patterns [4–7]. No effective algorithm has been suggested up till now to generate tests of size close to theoretical bounds (1). Though the number of operations in the algorithm suggested is still too large for practical applications, the approach developed in this paper seems to be promising and allows further improvements.

References

[1] D.T. Kleitman and J. Spencer, Families of k-independent sets, Discrete Math. 6 (1973) 255–262.
[2] N. Alon, Explicit construction of exponential sized families of k-independent sets, Discrete Math 58 (1986) 191–193.
[3] L.B. Levitin and M.G. Karpovsky, Efficient exhaustive test based on MDS codes, IEEE Internat. Symp. Inform. Theory, Ann Arbor, MI, USA (1986).

[4] A.K. Chandra, L.T. Kou, G. Markowsky and S. Zaks, On sets of Boolean n-vectors with all k-projections surjective, IBM Research Report, RC-8936 (1981).

[5] Z. Barzilai, D. Coppersmith and A. Rosenberg, Exhaustive bit pattern generation in discontinuous positions, with applications to VLSI self-testing, IBM Research Report RC-8750 (1981).

[6] D.T. Tang and C.L. Chen, Efficient exhaustive pattern generation for logic design, IBM Research Report RC-10064 (1983).

[7] G. Cohen, M. Karpovsky and L. Levitin, Exhaustive testing of circuits with outputs depending on limited number of inputs, IEEE Internat. Information Workshop, Caesarea, Israel (1984).